IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

IN RE PATENT APPLICATION OF:	Paul Haverstock, *et al.*
SERIAL NO.:	09/100,129
FILING DATE:	June 19, 1998
ATTORNEY DOCKET NO.:	042846-0313278 (*23452-034*)
CONFIRMATION NO.:	8225
ART UNIT :	2141
EXAMINER	Paul H. Kang
FOR:	WEB SERVER WITH AUTOMATED WORK FLOW

APPELLANTS' BRIEF ON APPEAL UNDER 37 C.F.R. §41.37

Mail Stop Appeal Brief - Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA. 22313-1450

Dear Sir:

Further to the Notice of Appeal filed on **October 13, 2004**, and in response to the Notification of Non-Compliant Appeal Brief mailed **May 20, 2005**, Appellants respectfully submit a complete new Appeal Brief pursuant to 37 C.F.R. §41.37.

The $500.00 fee for filing an Appeal Brief was submitted with the Appeal Brief filed December 13, 2004, pursuant to 37 C.F.R. §41.20(b)(2), therefore no fees are believed to be due. However, the Director is hereby authorized to charge any additional fees that may be due, or credit any overpayment of same to Deposit Account No. 033975 **(Ref. No. 042846-0313278)**.

REQUIREMENTS OF 37 C.F.R. §41.37

I. REAL PARTY IN INTEREST

The real party in interest is International Business Machines Corporation.

II. RELATED APPEALS AND INTERFERENCES

Appellants are aware of no related appeals or interferences.

III. STATUS OF CLAIMS

Pending: Claims 1-4, 6-11, 13-19, 21-24, and 26-33 are pending.

Cancelled: Claims 5, 12, 20, and 25 are cancelled.

Rejected: Claims 1-4, 6-11, 13-19, 21-24, and 26-33 stand rejected.

Allowed: No claims have been allowed.

On Appeal: The rejections of claims 1-4, 6-11, 13-19, 21-24, and 26-33 under 35 U.S.C. § 103(a) are appealed.

IV. STATUS OF AMENDMENTS

No amendments have been filed subsequent to the Final Office Action (Paper No. 46) mailed April 15, 2004 (hereinafter the "4/15/2004 Final Office Action").

V. SUMMARY OF CLAIMED SUBJECT MATTER

According to an embodiment of the invention, a server is provided that can respond to requests from a web browser for either HTML or non-HTML documents and return the

requested documents to the web browser. A workflow module distributes, routes and tracks documents according to predetermined processes.

In one embodiment, a server (14) is provided which may comprise an HTTP server module (30), interface module (32), and a non-markup language server module (24). *See* Specification, *e.g.*, pg. 5, lines 18-20; and FIGS. 1-2. The recitation of "server means" may refer to *at least* server (14) as described above. Similarly, the recitation of "HTTP server module means" may refer to *at least* HTTP server module (30) as described above, while the recitation of "non-HTML server module means" may refer to *at least* non-markup language server module (24) as described above.

In one embodiment, one or more databases may be in communication with server (14). For example, an HTML database (48) in communication with server (14) may comprise HTML objects (50a-50n). A non-HTML database (16) in communication with server (14) may store one or more non-HTML objects (18a-18n), at least some of which having one or more non-HTML fields (62a-62n), and a user directory (20). As such, server (14) enables a browser (28) to request both HTML objects (50a-50n) and non-HTML objects (18a-18n). *See* Specification, *e.g.*, pg. 5, lines 11 – pg. 6, line 15; and FIG. 1. The recitation of "database means" may refer to either one or both of *at least* HTML database (48) and non-HTML database (16) as described above.

In one embodiment, as illustrated in FIG. 2, server (14) may further comprise a workflow module (38). Workflow module (38) may automate tasks associated with transferring documents within a system. *See* Specification, *e.g.*, pg. 5, lines 1-3; pg. 9, lines 14-21; and FIG. 2.

Workflow module (38) may facilitate one or more object management tasks of server (14), associated with the one or more non-markup language objects (18a-18n)

according to one or more predefined calendaring and scheduling functions. For instance, workflow module (38) may distribute, route, and track documents according to a predefined process. *See* Specification, *e.g.*, pg. 9, lines 14-15. A schedule and calendar module (36) enables calendaring and scheduling functionality which provides developers with the tools to create web applications incorporating workflow. Business processes that have time-sensitive actions and tasks associated with multiple users can be rapidly enabled. For example, a web application can intelligently route messages and forms to users based on a variety of business relevant conditions (*e.g.*, user, schedule, time, and priority). Also, a web application can be designed to examine a group of users' calendars, determine the optimal schedule to ensure business processes are completed, and notify each user of impending work. *See* Specification, *e.g.*, pg. 15, line 18 – pg. 16, line 14; and FIG. 2.

Workflow module (38) may also notify at least one user that at least one action is required for the one or more non-markup language objects. For example, workflow module (38) may route documents to specific system users in a predetermined order. If a document must be reviewed by specific system users, workflow module (38) routes the document to the system users according to a routing process. If a system user does not act on the document before a predetermined time, workflow module (38) may notify the system user that action is required, forward the document to another system user, or perform another action identified in the routing process. *See* Specification, *e.g.*, pg. 9, lines 14-21. The recitation of "workflow means" may refer to *at least* workflow module (38) as described above.

In one embodiment, a markup language translator is provided that translates one or more non-markup language objects to representations of one or more markup language

objects so as to enable users to execute at least one action notified by workflow module (38). For example, HTML translator (44) may translate a non-HTML object to an HTML representation of a requested object, wherein the HTML representation is returned to browser (28). *See* Specification, *e.g.*, pg. 6, lines 9-11; and FIG. 1. The recitation of "markup language translating means" may refer to *at least* HTML translator (44).

VI. GROUNDS OF REJECTION NOT TO BE REVIEWED ON APPEAL (Double Patenting)

Appellants note the rejection of claims 1-4, 6-11, 13-19, 21-24, and 26-33 under the judicially created doctrine of obviousness-type double patenting over claims 1-20 of U.S. Patent No. 6,064,977 in view of U.S. Patent No. 6,073,109 to Flores *et al.* *See* 4/15/2004 Final Office Action, pg. 2, ¶3. Since the ultimate determination for filing a terminal disclaimer is based on the subject matter of the allowed claim and no claims have been indicated to be allowed in the present application, Appellants are not presenting this rejection for review on appeal. Rather, upon receiving a favorable outcome from the Board of Patent Appeals and Interferences on the 35 U.S.C. § 103 rejection, Appellants will consider filing a terminal disclaimer to place the application in condition for allowance.

VII. GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL (35 U.S.C. § 103).

Claims 1-4, 6-11, 13-19, 21-24, and 26-33 stand rejected under 35 U.S.C. §103(a) as allegedly being unpatentable over U.S. Patent No. 6,073,109 to Flores *et al.* (hereinafter "Flores") in view of U.S. Patent No. 5,745,360 to Leone. *See* 4/15/2004 Final Office Action, pg. 4, ¶5.

VIII. ARGUMENT

Claims 1-4, 6-11, 13-19, 21-24, and 26-33 are patentable for *at least* the reason that the Examiner has failed to set forth a *prima facie* case of obviousness under 35 U.S.C. 103(a).

To establish a *prima facie* case of obviousness, three basic criteria must be met. First, there must be some suggestion or motivation, either in the references themselves or in the knowledge generally available to one of ordinary skill in the art, to modify the reference or to combine reference teachings. Second, there must be a reasonable expectation of success. Finally, the prior art reference (or references when combined) must teach or suggest all the claim limitations. The teaching or suggestion to make the claimed combination and the reasonable expectation of success must both be found in the prior art, and not based on applicant's disclosure. *In re Vaeck*, 947 F.2d 488, 20 USPQ2d 1438 (Fed. Cir. 1991).

A. No Legally Proper Suggestion or Motivation to Combine Flores and Leone.

Independent claims 1, 8, 15, and 21 each generally recite, *inter alia*, the claim element of translating the one or more non-markup language objects to representations of one or more markup language objects in order to enable said at least one user to execute said at least one action notified by the workflow process.

In the 4/15/2004 Final Office Action, at pg. 5, the Examiner concedes that Flores fails to teach this claim element:

> However Flores does not explicitly teach the system, method and electronic storage medium further comprising a markup language translator that translates the one or more non-markup language objects to representations of one or more markup language objects in order to enable

> said at least one user to execute said at least one action notified by the workflow module.

The Examiner, however, relies on Leone for this claim element:

> In the same field of endeavor, Leone teaches a dynamic hypertext link converter system wherein non-hypertext documents are translated into hypertext documents in order to provide access to legacy databases (See Leone, col. 1, line 5 – col. 2, line 19).
>
> *See* 4/15/2004 Final Office Action, at pg. 5.

The Examiner then recites his suggestion or motivation for modifying Flores to include the teaching of Leone:

> It would have been obvious to one having ordinary skill in the art at the time the invention was made to have incorporated the system, method and electronic storage medium for translating a non-markup language object to a markup language object as taught by Leone, into the workflow system of Flores, for the purpose of enhancing interoperability of legacy systems by enabling operation over the internet.
>
> *See* 4/15/2004 Final Office Action, at pg. 5.

Neither Flores nor Leone provide support for the Examiner's alleged motivation to combine the two references. Neither of these two references even mentions any legacy system. Flores is concerned with a conventional workflow process, whereas Leone is concerned with translating electronic text documents, specifically book documents, to HTML.

Additionally, the Examiner has failed to set forth why one of ordinary skill in the art would even be motivated to modify Flores' workflow process to translate the workflow objects to a markup language. Accordingly, neither Flores, nor Leone, set forth any teaching, suggestion, or motivation to combine the two references. Accordingly, for at least the reasons set forth above, the Examiner has failed to set forth a *prima facie* case of obviousness.

B. Flores and Leone Fail to Teach or Suggest all the Claim Elements.

Assuming arguendo that there was a teaching, suggestion, or motivation to combine the two references, the rejection would still be improper as Flores and Leone, even when combined, fail to disclose, teach or suggest all of the elements of independent claims 1, 8, 15, and 21.

As recited above, the Examiner concedes that Flores "*...does not explicitly teach the system, method and electronic storage medium further comprising a markup language translator that translates the one or more non-markup language objects to representations of one or more markup language objects in order to enable said at least one user to execute said at least one action notified by the workflow module.*" The Examiner's reliance on Leone for this claim element, however, is misplaced. Neither Flores nor Leone, either alone or in combination, disclose *at least* this claim element.

Flores appears to disclose a workflow enabled system for facilitating business processes. At best, Flores describes notifying a user of steps to be completed and managing reminders to the user to keep the process of completing the task moving. Flores does not, however, teach or suggest translating a non-markup language object in a workflow system to its representation of a markup language object so that a user of a browser can execute a workflow action required for the translated non-markup language object using the browser. Leone does not remedy this deficiency.

Leone appears to describe a hypertext converter system that converts a non-HTML document to an HTML document. A document that is not formatted as HTML may be converted to an HTML document so that the document may be accessed over

communications networks. Leone does not teach or suggest enabling a user to execute a workflow action required for a translated non-markup language object using the browser in response to translating the non-markup language object to a markup language object, and as such, does not cure the deficiencies of Flores.

Accordingly, independent claims 1, 8, 15, and 21 are patentable for *at least* the reason that the Examiner has failed to set forth a *prima facie* case of obviousness under 35 U.S.C. 103(a). Dependent claims 2-4, 6-7, 9-11, 13-14, 16-19, 22-24, and 26-33 are allowable because they depend from allowable independent claims for the reasons set forth above, as well as for the further limitations they contain.

Claims 6 and 13.

Dependent claim 6 further recites the claim element of "wherein the server comprises a HTTP server module." Dependent claim 13 further recites the claim element of "wherein the server means comprises a HTTP server module means for communicating with one or more markup language database means."

In the 4/15/2004 Final Office Action (at pg. 4, ¶6), the Examiner first alleges that the server is taught by Flores at FIG. 2, and col. 8, lines 34-43. With regard to dependent claims 6 and 13, the Examiner alleges that the claimed HTTP server module is taught by Leone at FIG. 2, and col. 4, lines 6-45. *See* 4/15/2004 Final Office Action, at pg. 6, ¶10.

The passage of Flores relied upon by the Examiner as allegedly teaching a server includes references to a "workflow server," but does not teach an HTTP server module. While the passage of Leone cited by the Examiner refers to an HTTPD server, the Examiner has failed to provide any suggestion or motivation for modifying the workflow server of Flores to be an HTTPD server module. As such, the Examiner has failed to set

forth a *prima facie* case of obviousness. For *at least* this reason, the rejection of claims 6 and 13 is improper, and should be withdrawn.

Claims 7 and 14.

Dependent claim 7 further recites the claim element of "wherein the server comprises a non-markup language server module." Dependent claim 14 further recites the claim element of "wherein the server means comprises non-HTML server module means for communicating with the non-markup language database means."

In the 4/15/2004 Final Office Action (at pg. 4, ¶6), the Examiner first alleges that the server is taught by Flores at FIG. 2, and col. 8, lines 34-43. With regard to dependent claims 7 and 14, the Examiner alleges that the claimed non-markup language server module is taught by Leone at col. 4, lines 6-45. *See* 4/15/2004 Final Office Action, at pg. 6, ¶11.

The passage of Flores relied upon by the Examiner as allegedly teaching a server includes references to a "workflow server," but does not teach a non-markup language server module. The passage of Leone relied upon by the Examiner does not appear to teach a non-markup language server module and thus, does not remedy the deficiency of Flores.

Assuming arguendo that Leone did teach a non-markup language server module, the rejection would still be improper as the Examiner has failed to provide any suggestion or motivation for modifying the workflow server of Flores to be a non-markup language server module. As such, the Examiner has failed to set forth a *prima facie* case of obviousness. For *at least* this reason, the rejection of claims 7 and 14 is improper, and should be withdrawn.

IX. APPENDIX

The pending claims (claims 1-4, 6-11, 13-19, 21-24, and 26-33) are attached in the Appendix.

CONCLUSION

For at least the foregoing reasons, Appellant respectfully requests that the rejection of each of pending claims 1-4, 6-11, 13-19, 21-24, and 26-33 under 35 U.S.C. §103(a) be reversed.

Date: **June 20, 2005**

Respectfully submitted,

By: # 48,283
for Rick A. Toering
Registration No. 43,195

Customer No. 00909

Pillsbury Winthrop Shaw Pittman LLP
P.O. Box 10500
McLean, Virginia 22102
Main: 703-905-2000
Direct Dial: 703-905-2141
Fax: 703-905-2500

APPENDIX

1. **(Previously Presented)** A server system facilitating one or more object management tasks, of a server, associated with one or more non-markup language objects, the system comprising:

a server;

one or more databases, in communication with the server, comprising one or more non-markup language objects;

a workflow module that facilitates one or more object management tasks of the server, associated with the one or more non-markup language objects according to one or more predefined calendaring and scheduling functions, wherein the workflow module notifies at least one user that at least one action is required for the one or more non-markup language objects; and

a markup language translator that translates the one or more non-markup language objects to representations of one or more markup language objects in order to enable said at least one user to execute said at least one action notified by the workflow module.

2. **(Previously Presented)** The system of claim 1, wherein the workflow module performs the one or more tasks based on an occurrence of one or more events.

3. **(Previously Presented)** The system of claim 2, wherein the workflow module distributes the one or more objects according to the predefined process.

4. **(Previously Presented)** The system of claim 1, further comprising a notifying module that notifies the system user that an action is required for the one or more objects.

5. **(Cancelled)**

6. **(Previously Presented)** The system of claim 1, wherein the server comprises a HTTP server module.

7. **(Previously Presented)** The system of claim 1, wherein the server comprises a non-markup language server module.

8. **(Previously Presented)** A server system facilitating one or more object management tasks, of a server, associated with one or more non-markup language objects, the system comprising:

server means;

database means, in communication with the server means, for storing one or more non-markup language objects;

workflow means for facilitating one or more object management tasks of the server, the workflow means associated with the one or more non-markup language objects according to one or more predefined calendaring and scheduling functions, wherein the workflow means notifies at least one user that at least one action is required for the one or more non-markup language objects; and

markup language translating means for translating the one or more non-markup language objects to representations of one or more markup language objects in order to enable said at least one user to execute said at least one action notified by the workflow means.

9. **(Previously Presented)** The system of claim 8, wherein the workflow means performs the one or more tasks based on an occurrence of one or more events.

10. **(Previously Presented)** The system of claim 8, wherein the workflow means distributes the one or more non-markup language objects according to the predefined process.

11. **(Previously Presented)** The system of claim 10, further comprising notifying means for notifying a system user that an action is required for the one or more objects.

12. **(Cancelled)**

13. **(Previously Presented)** The system of claim 8, wherein the server means comprises a HTTP server module means for communicating with one or more markup language database means.

14. **(Previously Presented)** The system of claim 8, wherein the server means comprises non-HTML server module means for communicating with the non-markup language database means.

15. **(Previously Presented)** A method for facilitating one or more object management tasks, of a server, associated with one or more non-markup language objects, the method comprising the steps of:

storing one or more non-markup language objects in one or more databases;

creating a workflow process to be applied to the one or more non-markup language objects, wherein the work flow process notifies at least one user that at least one action is required for the one or more non-markup language objects;

determining a workflow process to apply to one or more non-markup language objects;

applying the workflow process to the one or more non-markup language objects, wherein the workflow process performs one or more object management tasks relating to the one or more non-markup language objects according to one or more predefined calendaring and scheduling functions; and

translating the one or more non-markup language objects to representations of one or more markup language objects in order to enable said at least one user to execute said at least one action notified by the workflow process.

16. **(Previously Presented)** The method of claim 15, further comprising the step of applying the workflow process according to a predetermined process.

17. **(Previously Presented)** The method of claim 15, further comprising the step of distributing the one or more non-markup language objects according to the workflow process.

18. **(Previously Presented)** The method of claim 17, further comprising the step of notifying a system user that an action is required for the one or more objects.

19. **(Previously Presented)** The method of claim 15, further comprising the step of performing the workflow process based on an occurrence of one or more events.

20. **(Cancelled)**.

21. **(Previously Presented)** An electronic storage medium having code embodied therein for causing a processor to facilitate one or more object management tasks, of a server, associated with one or more non-markup language objects, the medium comprising:

communicating code that causes a processor to enable a server to communicate with one or more databases comprising one or more non-markup language objects; and

workflow facilitating code that causes a processor to facilitate workflow of one or more object management tasks of the server, associated with the one or more non-markup language objects according to one or more predefined calendaring and scheduling functions, wherein the workflow process notifies at least one user that at least one action is required for the one or more non-markup language objects; and

markup language translating code that causes a processor to facilitate translating the one or more non-markup language objects to representations of one or more markup language objects in order to enable said at least one user to execute said at least one action notified by the workflow process.

22. **(Previously Presented)** The medium of claim 21, wherein the workflow facilitating code performs the one or more tasks based on an occurrence of one or more events.

23. **(Previously Presented)** The medium of claim 2 1, wherein the workflow facilitating code distributes the one or more non-markup language objects according to the predefined process.

24. **(Previously Presented)** The medium of claim 23, further comprising notifying code that causes a processor to notify a system user that an action is required for the one or more non-markup language objects.

25. **(Cancelled)**.

26. **(Previously Presented)** The system of claim 1, wherein the workflow module routes the one or more objects according to the predefined process.

27. **(Previously Presented)** The system of claim 1, wherein the workflow module tracks the one or more objects according to the predefined process.

28. **(Previously Presented)** The system of claim 8, wherein the workflow means routes the one or more objects according to the predefined process.

29. **(Previously Presented)** The system of claim 8, wherein the workflow means tracks the one or more objects according to the predefined process.

30. **(Previously Presented)** The method of claim 15, further comprising the step of routing the one or more non-markup language objects according to the workflow process.

31. **(Previously Presented)** The method of claim 15, further comprising the step of tracking the one or more non-markup language objects according to the workflow process.

32. **(Previously Presented)** The medium of claim 21, wherein the work flow facilitating codes routes the one or more objects according to a predefined process.

33. **(Previously Presented)** The medium of claim 21, wherein the work flow facilitating code tracts the one or more objects according to the predefined process.